PROCESS GUI Statement Actions

General Information

The PROCESS GUI statement actions execute procedures from within the PROCESS GUI statement. They are flexible in parameter handling: it is sufficient to supply type-compatible parameters with them (for example, if A5 is the exact parameter type, supplying an I1 parameter will be accepted).

For more information about the PROCESS GUI statement, refer to **Executing Standardized Procedures** in the section Event-Driven Programming Techniques of your Natural User's Guide.

To use the PROCESS GUI statement actions more comfortably, the local data area NGULKEY1 is automatically included in the list of local data areas used by any new dialog.

NGULKEY1 contains reserved symbols to be used in any event handler code. This enables you to use meaningful names as parameters in a PROCESS GUI statement. It also enables you to refer to certain attribute values by the more meaningful texts rather than by the integer values.

Note:
The "Response" parameters in the PROCESS GUI statement actions are optional.

Note:
ActiveX controls use the PROCESS GUI statement in a slightly different way. You use the statement to execute the ActiveX control's own methods and to access parameterized properties. For a description on how to use the PROCESS GUI statement for ActiveX controls, see your Natural User's Guide - Working with ActiveX Controls.

ADD Action

Description

Creates a single specified dialog element dynamically. See also **Creating/Deleting Dialog Elements Dynamically** in the section Event-Driven Programming Techniques of your Natural User's Guide. This action is most frequently used to add items to a list box control or to a selection-box control or to add column specifications to a table. It can also be used to create all kinds of dialog elements dynamically.

There are two syntax options of this action:

PROCESS GUI ACTION ADD WITH ...

This option has the parameters as listed below. Other attributes of the newly created dialog element have to be set in the global attributes list before the PROCESS GUI statement.

PROCESS GUI ACTION ADD WITH PARAMETERS ... END-PARAMETERS

This option accepts a list of attribute assignments, one for each attribute that is to be specified for the newly created dialog element. If you use this option, the global attribute list is not used or affected. For all attributes that are not explicitly specified, the default value is taken.

Parameters for the ADD WITH option

Name/Data Type	Explanation
HANDLE OF GUI	Input The handle of the parent dialog element.
Type (I4)	Input The type of dialog element to be created.
HANDLE OF GUI	Output The handle of the newly created dialog element.
Response (I4)	Output Natural error (if applicable).

Example 1 (option 1):

```
DEFINE DATA LOCAL
1 #NEW1 HANDLE OF INPUTFIELD
END-DEFINE
...
#NEW1.STRING:= 'NEW1'
#NEW1.RECTANGLE-X:= 24
#NEW1.RECTANGLE-Y:= 30
#NEW1.RECTANGLE-W:= 176
#NEW1.RECTANGLE-H:= 28
#NEW1.ENABLED:= TRUE
#NEW1.VISIBLE:= TRUE
PROCESS GUI ACTION ADD WITH #DLG$WINDOW INPUTFIELD #NEW1
```

Example 2 (option 2):

```
DEFINE DATA LOCAL
1 #NEW2 HANDLE OF INPUTFIELD
END-DEFINE
...
PROCESS GUI ACTION ADD WITH PARAMETERS
   HANDLE-VARIABLE = #NEW2
   TYPE = INPUTFIELD
   STRING = 'NEW2'
   RECTANGLE-X = 24
   RECTANGLE-Y = 30
   RECTANGLE-W = 176
   RECTANGLE-H = 28
   ENABLED = TRUE
   VISIBLE = TRUE
   PARENT = #DLG$WINDOW
END-PARAMETERS
```

If you insert a new dialog element dynamically by using the ADD action, you determine its position in the navigation sequence by creating the dialog element and setting the SUCCESSOR attribute to the handle value of its successor.

Example:


```
/* Insert input-field control #NEW1 before push button control #PB-1
/* Be careful not to trigger the PROCESS GUI statement action from a push
/* button control named #PB-1 because you are already defining it
DEFINE DATA LOCAL
1 #NEW1 HANDLE OF INPUTFIELD
1 #PB-1 HANDLE OF PUSHBUTTON
END-DEFINE
...
PROCESS GUI ACTION ADD WITH PARAMETERS
   PARENT = #DLG$WINDOW
   HANDLE-VARIABLE = #NEW1
   TYPE = INPUTFIELD
   SUCCESSOR = #PB-1
   ...
END-PARAMETERS
```

ADD-ITEMS Action

Description

Adds several list box items in a list box control at once. Does the same for selection-box items in a selection-box control and column specifications in a table control.

Parameters

Name/Data Type	Explanation
HANDLE OF dialog element	Input Specifies a dialog element.
Number of Items (I4)	Input You can add any number of items.
Item (list of A253-compatible values or one-dimensional array)	Input Item string(s).
Response (I4)	Output Natural error (if applicable).

Example:

```
DEFINE DATA LOCAL
1 #AMOUNT (I4)
1 #ITEM (A20/1:5)
1 #RESPONSE (I4)
END-DEFINE
...
#AMOUNT:= 5
#ITEM(1):= 'Berlin'
#ITEM(2):= 'Paris'
#ITEM(3):= 'London'
#ITEM(4):= 'Milan'
#ITEM(5):= 'Madrid'
PROCESS GUI ACTION ADD-ITEMS WITH #LB-1 #AMOUNT #ITEM(*) GIVING #RESPONSE
```


ADD-ITEMS-EX Action

Description

Adds list box items in a list box control and selection-box items in a selection-box control. Associates data to the items either as CLIENT-KEY/CLIENT-VALUE pairs or as CLIENT-DATA values. The item strings and the corresponding data can be specified either as single values or as one-dimensional arrays. Both strings and data must be provided in the same way. If the parameter "Client key" is blank, each data value is entered as the CLIENT-DATA of the corresponding item. If it is not blank, it is entered as the CLIENT-VALUE with the corresponding CLIENT-KEY.

Parameters

Name/Data Type	**Explanation**
HANDLE OF dialog element	Input Specifies a dialog element.
Number of Items (I4)	Input
Client key (A253)	Input CLIENT-KEY attribute to be used for each value in the list of values.
String list (A253)	Input A number of alphanumeric variables or constants or an array specification.These strings are entered in the STRING attribute of each item. You may only use a one-dimensional array in the array specification. Using part of a higher-level array causes an error.
Value list (A253 or I4)	Input A number of alphanumeric variables or constants or an array specification.These are interpreted as the CLIENT-VALUE of the given CLIENT-KEY. If the CLIENT-KEY value is blank, the value list is entered in the CLIENT-DATA.
Response (I4)	Output Natural error (if applicable).

Example:

```
/* Definitions in the dialog's local data area:
1 #NUMBER (N4)
1 #CITY (A20/1:2)
1 #CODE (I4/1:2)
1 #KEY (A20)
...
/* Event handler code:
#NUMBER:= 2
#CITY(1):= 'Berlin'
#CODE(1):= 1015
#CITY(2):= 'Munich'
#CODE(2):= 8053
#KEY:= 'Code'
PROCESS GUI ACTION ADD-ITEMS-EX
 WITH #LB-1 #NUMBER #KEY #CITY(1) #CITY(2) #CODE(1) #CODE(2) GIVING #RESPONSE
/*  Another possible WITH clause (same result)
 WITH #LB-1 #NUMBER #KEY #CITY(1:2) #CODE(1:2) GIVING #RESPONSE
```


BEEP Action

Description

Issues a "beep" sound.

Parameters

Name/Data Type	Explanation
Number (1 - 3)	Input Type of the beep to be issued. "1" denotes a warning beep, "2" denotes a note beep, "3" denotes an error beep.

Note:
The "Number" parameter is optional. If you omit this parameter, the value "1" (warning beep) applies by default.

Example:

```
   PROCESS GUI ACTION BEEP
```


CLEAR Action

Description

Clears the contents of an input-field control, an edit area control or a table control.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Specifies an input-field control, an edit area control or a table control.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION CLEAR WITH #EA-1 GIVING #RESPONSE
```


DELETE Action

Description

Deletes a specified dialog element. You can use this action, for example, to delete items from a list box control or to remove dialog elements from a dialog.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Any dialog element.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION DELETE WITH #PB-1 GIVING #RESPONSE
```

DELETE-CHILDREN Action

Description

Deletes all children of a given dialog or dialog element dynamically. You use this action, for example, to delete all items in a list box control before filling the list box again (using the ADD-ITEMS action).

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Handle of the dialog or dialog element whose children are to be deleted.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION DELETE-CHILDREN WITH #LB-1 GIVING #RESPONSE
```


DELETE-WINDOW Action

Description

Deletes (unloads) a dialog regardless of whether it has actually been created or not. You use this action, for example, to ensure that the dialog is unloaded if a CLOSE DIALOG statement is performed in the before-open event handler. Normally, you cannot close a dialog before it is actually created. To avoid a conflict, you use this action.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Handle of the window which is to be deleted.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION DELETE-WINDOW WITH #MYDIA GIVING #RESPONSE
```


EDIT-GET-LINE-NUMBER Action

Description

Retrieves the number of lines in the edit area control. This is also true if the STYLE attribute value has been set to value "w" (wordwrap).

Parameters

Name/Data Type	Explanation
HANDLE OF EDITAREA	Input Specifies an edit area control.
Line number (I4)	Output The number of lines.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION
EDIT-GET-LINE-NUMBER WITH #EA-1
#LINE-NUMBER GIVING
#RESPONSE
#IF-1.STRING:= #LINE-NUMBER
```


EDIT-LINE-DELETE Action

Description

Deletes a line in the edit area control.

Parameters

Name/Data Type	Explanation
HANDLE OF EDITAREA	Input Specifies an edit area control.
Line number (I4)	Output The line to be deleted. If you specify "0", the last line will be deleted.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION EDIT-LINE-DELETE WITH #EA-1
#LINE-NUMBER GIVING #RESPONSE
```

EDIT-LINE-GET-SELECTION Action

Description

Retrieves the extent of the selected area (lines and columns) in an edit area control.

Parameters

Name/Data Type	Explanation
HANDLE OF EDITAREA	Input Specifies an edit area control.
Line from (I4)	Output Selection starts from this line onwards.
Column from (I4)	Output Selection starts from this column onwards.
Line to (I4)	Output Last selected line.
Column to (I4)	Output The column position immediately following the last selected character.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION
EDIT-LINE-GET-SELECTION WITH #EA-1
#LINE-FROM #COLUMN-FROM
#LINE-TO #COLUMN-TO GIVING #RESPONSE
```


EDIT-LINE-GET-TEXT Action

Description

Retrieves text in a line of the edit area control.

Parameters

Name/Data Type	Explanation
HANDLE OF EDITAREA	Input Specifies an edit area control.
Line number (I4)	Input The text is retrieved out of this line.
Column from (I4)	Input The text is retrieved from this column onwards.
Column to (I4)	Input/Output The text is retrieved up to this column. If you specify "0", the complete line is retrieved. The position immediately following the last character in the line will be returned (or 1, if the line is empty).
Line text (A253)	Output Returns the retrieved text string.
Split (L)	Output Indicates whether the total length of the requested text exceeds the actually retrieved line text string.
Response (I4)	Output Natural error (if applicable).

Example:

```
#LINE-NUMBER := 1
#COLUMN-FROM := 1
#COLUMN-TO := 0
PROCESS GUI ACTION EDIT-LINE-GET-TEXT WITH #EA-1 #LINE-NUMBER #COLUMN-FROM
#COLUMN-TO #LINE-TEXT #SPLIT GIVING #RESPONSE
```


EDIT-LINE-INSERT Action

Description

Inserts a new line into an edit area control.

Parameters

Name/Data Type	Explanation
HANDLE OF EDITAREA	Input Specifies an edit area control.
Line number (I4)	Input/Output The new line will be inserted before this line. If you specify "0", the new line will be appended to the last line of the edit area control.
Line length (I4)	Input/Output The number of characters to be inserted into the new line starting from the first character of "Line text". If you specify "0", an empty line will be inserted. If you specify "-1", the "Line text" string will be copied into the new line, trailing blanks will be removed and the number of copied characters will be returned.
Line text (A253)	Input/Output The text string of the line to be inserted.
Response (I4)	Output Natural error (if applicable).

Example:

```
#LINE-NUMBER := 0
#LINE-LENGTH := 10
#LINE-TEXT := 'Hello!'
PROCESS GUI ACTION EDIT-LINE-INSERT WITH #EA-1 #LINE-NUMBER #LINE-LENGTH
#LINE-TEXT GIVING #RESPONSE
```


EDIT-LINE-SET-SELECTION Action

Description

Selects a range of text (lines, columns) in an edit area control. To set the text caret to a certain position, "Line from" and "Line to" must have the same value; "Column from" and "Column to" must also have the same value.

Parameters

Name/Data Type	Explanation
HANDLE OF EDITAREA	Input Specifies an edit area control.
Line from (I4)	Input/Output Selection starts from this line onwards. If you specify "0", the last line is selected.
Column from (I4)	Input/Output Selection starts from this column onwards. If you specify "0", the end of the line is selected as the starting column.
Line to (I4)	Input/Output Last selected line. If you specify "0", the last line is selected. The number of this last line will be returned.
Column to (I4)	Input/Output Last selected column. If you specify "0", the complete line is selected. The last selected position in the line will be returned.
Response (I4)	Output Natural error (if applicable).

Example:

```
#LINE-FROM := 1         /* Select the first three lines
#LINE-TO := 3
#COLUMN-FROM := 1       /* Select from the first to the last column
#COLUMN-TO := 0
PROCESS GUI ACTION EDIT-LINE-SET-SELECTION WITH #EA-1 #LINE-FROM #COLUMN-FROM
#LINE-TO #COLUMN-TO GIVING #RESPONSE
```


EDIT-LINE-SET-TEXT Action

Description

Replaces text in a line of the edit area control. Note that you can also use this action to

- insert text by setting parameters "Column from" and "Column to" to the same value (this will be the insertion position);
  or to
- delete text by setting the parameter "Line length" to the value ’0’.

Parameters

Name/Data Type	Explanation
HANDLE OF EDITAREA	Input Specifies an edit area control.
Line number (I4)	Input/Output The text is replaced out of this line.
Column from (I4)	Input/Output The text is replaced from this column onwards.
Column to (I4)	Input/Output The text is replaced up to this column. If you specify "0", the rest of the line is replaced. The last position in the line will be returned.
Line text (A253)	Input/Output Contains the string replacing the string specified with "Line number", "Column from", "Column to".
Line length (I4)	Input/Output The number of "Line text" characters (out of 253) that replace the old string starting from the first character. If you specify "-1", trailing blanks will be removed and the number of characters used will be returned.
Response (I4)	Output Natural error (if applicable).

Example:

```
#LINE-NUMBER := 1       /*Replaces line 1
#COLUMN-FROM := 1
#COLUMN-TO := 0
#LINE-TEXT := ’New text’
PROCESS GUI ACTION EDIT-LINE-SET-TEXT WITH #EA-1 #LINE-NUMBER #COLUMN-FROM
#COLUMN-TO #LINE-TEXT #LINE-LENGTH GIVING #RESPONSE
MOVE #LINE-TEXT TO #EA-1.STRING
```


GET-FOCUS Action

Description

Gets the handle of the dialog element which currently has the focus, that is, which responds to keyboard input.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Output The dialog element that has the focus.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION GET-FOCUS WITH #FOCUS GIVING #RESPONSE
```

GET-MESSAGE-TEXT Action

Description

Gets the text for a given application message file number.

Parameters

Name/Data Type	Explanation
Message number (I4)	Input The message number for which the text is to be read.
Destination (A253)	Output The message text.
Response (I4)	Output Natural error (if applicable).

Example:

```
   PROCESS GUI ACTION GET-MESSAGE-TEXT WITH #MESSAGENR #DEST GIVING #RESPONSE
```


HELP Action

Description

Invokes the help system and points to the given help topic in the help file HELP-FILENAME determined by the attribute for the specified dialog, or "*libraryname*.hlp" if this has not been set. Natural expects the help file to be located in one of the directories listed in the desription of the HELP-FILENAME attribute.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input The dialog or dialog element acting as parent window for the help window. Must not be an item. This parameter is also used to retrieve any non-default help file name via dialogs HELP-FILENAME attribute.
Help ID (I4)	Input The help topic ID.
Response (I4)	Output Natural error (if applicable).

Example:

```
#EA-1.HELP-ID := 1234
/*Set 1234 as help topic ID for the edit area
PROCESS GUI ACTION HELP WITH #EA-1 #EA-1.HELP-ID GIVING #RESPONSE
```


HOURGLASS-REMOVE Action

Description

Sets the pointer shape from "hourglass" to "arrow" and clears the pointer stack. You can also use this action to suspend the hourglass pointer and use the arrow shape while displaying a message box during long periods of processing. You resume with the hourglass pointer on the same nesting level as when it was suspended by passing the value returned in the level parameter to the HOURGLASS-STACK action.

Parameters

Name/Data Type	Explanation
Level (I4)	Output The pointer stack level before the stack is cleared.

Example:

```
   PROCESS GUI ACTION HOURGLASS-REMOVE WITH #LEVEL GIVING #RESPONSE
```


HOURGLASS-STACK Action

Description

Sets the pointer shape to "hourglass" and keeps the previous pointer shape in the stack of pointers. This indicates that end-user input is disabled during a long period of processing. To restore the previous shape (hourglass or arrow), you use the HOURGLASS-UNSTACK action. This stack/unstack logic can be used in nested program structures to determine the pointer shape.

The level parameter can be used to resume the hourglass on the same stack level as when it was suspended by the HOURGLASS-REMOVE action.

Note:
While a Natural dialog is performing a lengthy operation during which it is not able to process events, it should display the hourglass to the user to indicate that it is currently not available for operation. While the hourglass is active, no events will be delivered to the dialog.

Parameters

Name/Data Type	Explanation
Level (I4)	Output The pointer stack level (max. 10) up to which the stack is filled with hourglass pointers.

Note:
The "Level" parameter is optional.

Example:

```
PROCESS GUI ACTION HOURGLASS-STACK WITH #LEVEL GIVING #RESPONSE
```


HOURGLASS-UNSTACK Action

Description

Restores the previous pointer shape (hourglass or arrow) by decreasing the stack of pointers by one and by setting the pointer to the new stack top. If the stack is empty, the arrow shape is activated. This action corresponds to action HOURGLASS-STACK.

Parameters

Name/Data Type	Explanation
Level (I4)	Output The current pointer stack level.

Note:
The "Level" parameter is optional.

Example:

```
PROCESS GUI ACTION HOURGLASS-UNSTACK WITH #LEVEL GIVING #RESPONSE
```


INPUT-COPY-SELECTION Action

Description

The text currently selected in the input dialog element is copied to the clipboard. The text can be selected in an input-field control, a selection-box control or an edit area control.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Specifies an input-field control, a selection-box control or an edit area control.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION INPUT-COPY-SELECTION WITH #IF-1 GIVING #RESPONSE
```

Note:
For selection-boxes, the action must be called whilst the selection-box still has the focus.

INPUT-CUT-SELECTION Action

Description

The text currently selected in the input dialog element is copied to the clipboard and is deleted in the original place. The text can be selected in an input-field control, a selection-box control or an edit area control.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Specifies an input-field control, a selection-box control or an edit area control.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION INPUT-CUT-SELECTION WITH #IF-1 GIVING #RESPONSE
```

Note:
For selection-boxes, the action must be called whilst the selection-box still has the focus.

INPUT-DELETE-SELECTION Action

Description

The text currently selected in the input dialog element is deleted. The text can be deleted from an input-field control, a selection-box control or an edit area control.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Specifies an input-field control, a selection-box control or an edit area control.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION INPUT-DELETE-SELECTION WITH #IF-1 GIVING #RESPONSE
```

Note:
For selection-boxes, the action must be called whilst the selection-box still has the focus.

INPUT-GET-LINE-LENGTH Action

Description

Gets the length of a line in an edit area control or in an input-field control.

Parameters

Name/Data Type	Explanation
HANDLE OF EDITAREA or INPUTFIELD	Input Specifies an edit area control or an input-field control.
Line number (I4)	Input Length is retrieved from this line.
Line length (I4)	Output Returns the line length.
Response (I4)	Output Natural error (if applicable).

Example:

```
#LINE-NUMBER := 1   /* Examine the first line
PROCESS GUI ACTION INPUT-GET-LINE-LENGTH WITH #EA-1 #LINE-NUMBER #LINE-LENGTH
GIVING #RESPONSE
#IF-1.STRING := #LINE-LENGTH /* Display the result in this input-field control
```


INPUT-GET-SELECTION Action

Description

Retrieves the position of the selected text in an edit area control or in an input-field control from the beginning (character number) to the end (character number).

Parameters

Name/Data Type	Explanation
HANDLE OF EDITAREA or INPUTFIELD	Input Specifies an edit area control or an input-field control.
Position from (I4)	Output Selection starts from this position onwards.
Position to (I4)	Output Last selected position.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION INPUT-GET-SELECTION WITH #EA-1 #POSITION-FROM #POSITION-TO
GIVING #RESPONSE
#IF-1.STRING := #POSITION-FROM /* Display the result in these two
#IF-2.STRING := #POSITION-TO   /* input-field controls
```

INPUT-GET-TEXT Action

Description

Retrieves text in a line of the edit area control or in an input-field control (from a position onwards).

Parameters

Name/Data Type	Explanation
HANDLE OF EDITAREA or INPUTFIELD	Input Specifies an edit area control or an input-field control.
Position-from (I4)	Input The text is retrieved from this position onwards.
Line length (I4)	Input/Output The number of the characters to be retrieved (must be 1 to 253).
Line text (A253)	Output The retrieved text will be copied into this text string.
End (L)	Output Becomes TRUE if the last retrieved character was the last character in the dialog element. If "End" is TRUE, the number of retrieved characters is returned in the "Line length" parameter.
Response (I4)	Output Natural error (if applicable).

Example:

```
#POSITION := 8     /* Set input
#LINE-LENGTH := 22 /* values
PROCESS GUI ACTION INPUT-GET-TEXT WITH #EA-1 #POSITION-FROM #LINE-LENGTH
   #LINE-TEXT #END GIVING #RESPONSE
#IF-1.STRING := #POSITION    /* Display output values
#IF-2.STRING := #LINE-LENGTH /* in input-field controls
#IF-3.STRING := #TEXT
```


INPUT-PASTE Action

Description

Copies the content of the clipboard into an input-field control, a selection-box control, or an edit area control. If nothing is selected inside the dialog element, the clipboard text is inserted at the cursor position. If an area of text is selected, this selection is deleted and the content of the clipboard is inserted.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Specifies a selection-box control or an edit area control.
Response (I4)	Output Natural error (if applicable).

Example:

```
   PROCESS GUI ACTION INPUT-PASTE WITH #IF-1 GIVING #RESPONSE
```

INPUT-SET-SELECTION Action

Description

Selects an area (position to position) in an edit area control or in an input-field control.

Parameters

Name/Data Type	Explanation
HANDLE OF EDITAREA or INPUTFIELD	Input Specifies an edit area control or an input-field control.
Position from (I4)	Input Selection starts from this position onwards.
Position to (I4)	Input Last selected position. If you specify "0", the selection will extend to the last character.
Response (I4)	Output Natural error (if applicable).

Example:

```
#POSITION-FROM := 10  /* Select from the 10th to the 22nd character
#POSITION-TO := 22
PROCESS GUI ACTION INPUT-SET-SELECTION WITH #EA-1 #POSITION-FROM #POSITION-TO
GIVING #RESPONSE
```


INPUT-SET-TEXT Action

Description

Replaces text in an edit area control or in an input-field control (from position to position).

Parameters

Name/Data Type	Explanation
HANDLE OF EDITAREA or INPUTFIELD	Input Specifies an edit area control or an input-field control.
Position-from (I4)	Input Position of the first character to be replaced. If you specify "0" the text is appended to the text already present in the dialog element.
Position-to (I4)	Input/Output Position of the last character to be replaced. If you specify "0", the index of the last character in the dialog element will be returned.
Text length (I4)	Input/Output The number of characters to be inserted before "Position-to" starting from the first character of "Line text". If you specify "-1", the "Line text" string will be inserted into the new line, trailing blanks will be removed and the number of copied characters will be returned.
Text (A253)	Input This text string replaces the old one.
Response (I4)	Output Natural error (if applicable).

Example:

```
#POSITION-FROM := 8
#POSITION-TO := 24
#TEXT-LENGTH := 16
#TEXT := 'Insert this text'
PROCESS GUI ACTION INPUT-SET-TEXT WITH #EA-1 #POSITION-FROM #POSITION-TO
#TEXT-LENGTH #TEXT GIVING #RESPONSE
```


INPUT-UNDO Action

Description

Undoes the last editing action in an input-field control, a selection-box control or an edit area control. Editing actions that can be undone are entering, copying, pasting, cutting and deleting text in a dialog element.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Specifies an input-field control, a selection-box control or an edit area control.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION INPUT-UNDO WITH #EA-1 GIVING #RESPONSE
```


INQ-CLICKPOSITION Action

Description

Returns the x-axis and the y-axis position where the end user has clicked (relative to the dialog element’s rectangle). These coordinates are also updated immediately before a context-menu’s before-open event is called. The application can then combine this call with a call to either INQ-ITEM-BY-POSITION or TABLE-INQUIRE-CELL to determine which list box item or table control cell was "right-clicked".

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Specifies a dialog element.
X-Position (I4)	Output The x-axis position in pixels.
Y-Position (I4)	Output The y-axis position in pixels.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION INQ-CLICKPOSITION WITH #BM-1 #X-POSITION #Y-POSITION
GIVING #RESPONSE
#IF-1.STRING := #X-POSITION  /* Display the coordinates in these two
#IF-2.STRING := #Y-POSITION  /* input-field controls
```

INQ-DRAG-DROP Action

Description

Can be used in a drag-drop event of a bitmap control to retrieve the position where the dropping occurred. The parameters X-Position and Y-Position return the coordinates of the position in pixels relative to the upper left corner of the target bitmap control. The Mode parameter indicates whether SHIFT and/or CTRL was pressed while the dropping occurred.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Output The handle of the dialog in which the bitmap control was dropped.
HANDLE OF GUI	Output The dragged bitmap control’s handle.
Mode (I4)	Output Possible values: 0 no key. 1 SHIFT key. 2 CTRL key. 3 SHIFT+CTRL keys.
X-Position (I4)	Output X-axis position on the target.
Y-Position (I4)	Output Y-axis position on the target.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION INQ-DRAG-DROP WITH #DIA-HANDLE #DIA-ELEMENT #MODE
#X-POSITION #Y-POSITION GIVING #RESPONSE
#IF-1.STRING := #X-POSITION  /* Display the coordinates in these two
#IF-2.STRING := #Y-POSITION  /* input-field controls
```


INQ-INNER-RECT Action

Description

Provides the width and the height of the client rectangle (in pixels) in a dialog window or dialog element. For dialogs, this is the area into which dialog elements can be placed. Dialog elements cannot be placed on the decoration, the title bar, a tool bar, a menu bar, a status bar, and the horizontal and vertical scroll bars. For dialog elements, this is the area excluding any frame components such as borders and scroll bars.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Handle of the dialog or dialog element that contains the client rectangle. If the handle value is NULL-HANDLE, the width and height of the screen is returned.
Width (I4)	Output The width of the client rectangle.
Height (I4)	Output The height of the client rectangle.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION INQ-INNER-RECT WITH #DLG$WINDOW #WIDTH #HEIGHT GIVING
#RESPONSE
#IF-1.STRING := #HEIGHT /* Display the width and height
#IF-2.STRING := #WIDTH
```

INQ-ITEM-BY-POSITION Action

Description

Returns the item at a given position within a control. This is particularly useful for finding out which list box item or status bar pane was clicked with the right mouse button before a context menu is displayed (see example below).

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Specifies a list box or status bar control.
X-Position (I4)	Input Specifies the x-axis position in pixels relative to the top-left hand corner of the control.
Y-Position (I4)	Input Specifies the y-axis position in pixels relative to the top-left hand corner of the control.
Item (HANDLE OF GUI)	Output The handle of the list box item or status bar pane at the specified position, or NULL-HANDLE if none.
Response (I4)	Output NATURAL error (if applicable).

Example:

```
/* Sample BEFORE-OPEN code for context menu:
PROCESS GUI ACTION INQ-CLICKPOSITION WITH
#LB-1 #X-POSITION #Y-POSITION GIVING #RESPONSE
PROCESS GUI ACTION INQ-ITEM-BY-POSITION WITH
#LB-1 #X-POSITION #Y-POSITION #LBITEM GIVING #RESPONSE
```


LOAD-LAYOUT Action

Description

Loads the specified dialog's control bar layout (if any) that was previously saved for the current user via the SAVE-LAYOUT action. If no such saved layout exists, this action has no effect.

Note that the bar types (dockable or fixed) should match those at the time the layout was saved, otherwise the results are unpredictable.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Handle of the dialog for which the control bar layout is to be loaded.
Profile name (A253)	Input (optional parameter) The name of the profile from which the information is to be loaded. If this parameter is not specified, the name of the library containing the dialog is used.
Section name (A253)	Input (optional parameter) The name of the profile section from which the information is to be loaded. If this parameter is not specified, the name of the dialog is used.

Example:

```
  PROCESS GUI ACTION LOAD-LAYOUT WITH #DLG$WINDOW GIVING #RESPONSE
```

MOVE-NAVIGATION-ITEMS Action

Description

Moves a range of controls to a new position in the control sequence. The control sequence within the range of controls being moved remains unchanged.

Parameters

Name/Data Type	Explanation
First control (HANDLE OF GUI)	Input Specifies the first control in the range of controls which is to be moved. If this parameter is specified as NULL-HANDLE, the first control in the control sequence is used.
Last control (HANDLE OF GUI	Input Specifies the last control in the range of controls which is to be moved. If this parameter is specified as NULL-HANDLE, the last control in the control sequence is used.
Position To (HANDLE OF GUI)	Input Specifies the control which is to immediately precede the moved control(s) in the new control sequence. This control must not be one of the controls being moved. If this parameter is specified as NULL-HANDLE, the controls are moved to the end of the control sequence. If this parameter is specified as the handle of the dialog itself, the controls are moved to the front of the control sequence.
Response (I4)	Output NATURAL error (if applicable).

Example:

```
/* Move the controls in the current control sequence starting from #PB-1 and ending with the last
/* control in the control sequence to follow #SB-1
PROCESS GUI ACTION MOVE-NAVIGATION-ITEMS WITH
      #PB-1 NULL-HANDLE #SB-1 GIVING #RESPONSE
```


OLE-ACTIVATE

Description

Activates the default action of an OLE server programmatically. To enable the end user to activate the OLE server, you set the OLE container control’s MODIFIABLE attribute to TRUE. The end user then activates the OLE server by using the right mouse button to click in the OLE container control’s rectangle; this shows the context menu, from which the end user can choose the appropriate command. The OLE server can also be activated by using the left mouse button to double-click in the OLE container control’s rectangle. If the menu item "Show Object Verbs" is provided, the end user can also use this to activate the server.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Specifies an OLE container control.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION OLE-ACTIVATE WITH #OCT-1 GIVING #RESPONSE
```


OLE-DEACTIVATE

Description

Deactivates an OLE server programmatically, that is, in-place editing is finished. The end user can also deactivate the server by pressing ESC or by clicking into the Natural dialog outside the OLE container control.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Specifies an OLE container control.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION OLE-DEACTIVATE WITH #OCT-1 GIVING #RESPONSE
```


OLE-GET-DATA

Description

Reads an embedded object into a Natural variable. It is recommended to define an array of variables that is large enough to hold the object: some objects are at least 10 KB in size. You then get the object by executing this procedure; the second parameter contains the size of the array. The procedure returns the real size being used and a Natural error code other than "0" is returned. To avoid getting this error code, you first query the current size of the object with the OBJECT-SIZE attribute.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Specifies an OLE container control.
Variable	Input Variable or (usually) an array of variables.
Size (I4)	Input Size of the variable or the array of variables provided.
Real size (I4)	Output Real size of the variable or the array of variables.
Response (I4)	Output Natural error (if applicable).

Example:

```
#CURRSIZE := #OCT-1.OBJECT-SIZE  /* How large is the object?
IF..  /* If the object is too large, do the following...
...   /* If not, execute the procedure
END-IF
PROCESS GUI ACTION OLE-GET-DATA WITH #OCT-1 #MYVARI (1:5) 15000 #REALSIZE GIVING
#RESPONSE
```


OLE-INSERT-OBJECT

Description

Provides the end user with a dialog box which allows him/her to select and start an OLE server application or an external object.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Specifies an OLE container control.
String (A253)	Input Dialog box caption.
Flag (I4)	Input For future use.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION OLE-INSERT-OBJECT WITH #MYCONTAINER 'My Caption' #FLAG GIVING
#RESPONSE
```


OLE-READ-FROM-FILE

Description

Reads an embedded object into the OLE container control.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Specifies an OLE container control.
File name (A253)	Input Specifies the name of an embedded object (in Natural, this is a ".neo" file)
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION OLE-READ-FROM-FILE WITH #MYCONTAINER 'c:\natgui\myobject.neo'
GIVING #RESPONSE
```

OLE-SAVE-TO-FILE

Description

Saves the currently embedded object to a file with the default extension ".neo" (meaning "Natural Embedded Object").

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Specifies an OLE container control.
File name (A253)	Input Specifies the name of an embedded object (in Natural, this is a ".neo" file)
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION OLE-SAVE-TO-FILE WITH #MYCONTAINER 'MYOBJECT.NEO' GIVING
#RESPONSE
```


OLE-SET-DATA

Description

Puts the content of a Natural variable into an embedded object. Corresponds with the OLE-GET-DATA action. You put the variable content into the object by executing this procedure; the second parameter contains the size of the object. It is recommended to define an array of variables that is large enough to hold the object: some objects are at least 10 KB in size.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Specifies an OLE container control.
Variable	Input Variable or (usually) an array of variables.
Size (I4)	Input Size of the object.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION OLE-SET-DATA WITH #OCT-1 #MYVARI (1:5) #REALSIZE GIVING
#RESPONSE
```

PICK-FILENAME Action

Description

Invokes a standard dialog box that allows the end user to pick the name of a file which is **optionally** assigned to a Natural work file number.

Parameters

Name/Data Type	Explanation
Title (A253)	Input The text displayed in the title bar of the dialog box.
Init (A253)	Input A string pattern (*[directoryname\][filename[.extension]]*) that determines the initial directory (if specified), together with either the files to choose from (if the *filter list* parameter is not specified) or the pre-selected file (if a filename and extension containing no wildcard characters (*,?) is used). If the pattern is an empty string, all files of the current directory are listed.
Work file number (I4)	Input The work file number to which the selected file name is to be assigned. If this parameter is "0", the file name is not assigned to a work file number.
File name (A253)	Output Returns the selected file name with the entire path name.
Dialog Type (I4)	Input (optional parameter) Determines type of file selection dialog displayed: 0 (default) = 'Open' dialog (new files allowed); 1 = 'Open' dialog (selected file must exist); 2 = 'Save' dialog.
Filter list (A253/*)	Input (optional parameter) A one-dimensional array containing pre-specified display filter definitions. The array does not need to be filled entirely. Each filter definition is represented by a *pair* of array elements. The first element of each pair is a textual description of the filter used for display in the file selection dialog. The second element of each pair is the associated display filter pattern. The display filter pattern may be complex, consisting of multiple components separated by a semi-colon.
Filter index (I4)	Input (optional parameter) The index of the filter to be initially used when the file selection dialog is first opened. 0 = first filter, 1 = second filter , and so on.
Response (I4)	Output Natural error (if applicable).

Example:

```
DEFINE DATA LOCAL
1 #RESPONSE (I4)
1 #MYTITLE (A253)
1 #SELECTNAME (A253)
1 #FILTER (A32/6) CONST
  <
    'Private resource files', '*.nr*',
    'Shared resource files',  '*.bmp;*.ico;*.hlp;*.neo;*.rpt',
    'All Files (*.*)',        '*.*'
  >
END-DEFINE
...
#MYTITLE := 'Example using implicit display filter'
PROCESS GUI ACTION PICK-FILENAME WITH #MYTITLE 'c:\*.*' 0
#SELECTNAME GIVING #RESPONSE
#IF-1.STRING := #SELECTNAME
*
#MYTITLE := 'Example using explicit display filter'
PROCESS GUI ACTION PICK-FILENAME WITH #MYTITLE 'd:\fuser\mylib\res\'
  0 #SELECTNAME 0 #FILTER(*) 1 GIVING #RESPONSE
#IF-2.STRING := #SELECTNAME
```


PLAY-SOUND Action

Description

Plays a tune from a ".WAV" audio file.

Note:
This action is not available under Windows NT.

Parameters

Name/Data Type	Explanation
Audio file name (.wav)	Input Either the full path name of the audio file or just the file name. If you specify just the file name, this file must be located in the directory referred to by the environment variable NATGUI_BMP. If you specify a path name, this must be the fully expanded path name including the drive and the path specifications.

Examples:

```
PROCESS GUI ACTION PLAY-SOUND WITH 'CHIMES.WAV'
PROCESS GUI ACTION PLAY-SOUND WITH #FILE GIVING #RESPONSE
```


PROCESS-EVENTS Action

Description

Processes all outstanding messages for the Natural application (raising any associated events) before returning. This is particularly useful for allowing the user to interrupt a long operation. Typically, you create a simple abort dialog of style ’modal’ with a ’Cancel’ push button, and call this action periodically whilst processing the long operation. If the user has clicked the ’Cancel’ push button in the meantime, the CLICK event for this button will be received during the call, which typically sets an abort flag that can be tested on return and handled accordingly.

Note:
This action must be used with care, because ALL user interactions with the application will be processed during the call. This includes clicks on tool bar or menu items, for instance, unless these are explicitly disabled, or if a modal dialog is currently on display (as in the above example).

Parameters

Response (I4)	Output Natural error (if applicable).

Example:

```
  PROCESS GUI ACTION PROCESS-EVENTS GIVING #RESPONSE
```

RECALC-LAYOUT Action

Description

Causes the control bar layout for the specified dialog to be recalculated, which may result in the bar positions and/or sizes being adjusted. The control bars include the newer tool-bar and status-bar controls, but not the traditional tool bar and status bar available in earlier Natural versions.

Normally, Natural implicity causes a layout recalculation where necessary (e.g. immediately before the after-open event, or if a bar is created or moved). However, in order to reduce the number of screen refreshes to a minimum, it may be that only a delayed implicit layout recalculation is performed. Since some operations, such as getting the dialog's client area via the INQ-INNER-RECT action, require that the bars are in their final positions, a preceding manually-triggered layout recalculation may be necessary in some cases. Note that this is not the case for all bars defined statically with the dialog editor, because Natural automatically does an immediate layout recalculation before calling the user's AFTER-OPEN event handler.

Note that an immediate layout recalculation will clear any outstanding delayed layout recalculation request. Also, multiple delayed layout recalculation requests are consolidated (i.e., do not cause multiple refreshes).

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Handle of the dialog whose bar layout is to be recalculated.
Mode (L)	Input (optional parameter) If true (default), an immediate layout recalculation is performed. If false, a delayed layout recalculation is performed.

Example:

```
PROCESS GUI ACTION RECALC-LAYOUT WITH #DLG$WINDOW GIVING #RESPONSE
```


REFRESH-LINKS Action

Description

Refreshes the content of those input-field controls and selection-box controls for which the linked variable option was chosen for the content variables. This becomes necessary after these variables have been changed in code to display new values, for example, after a new record has been read from a database.

Note:
If the content of such a dialog element is modified by the end user, the corresponding linked variable is updated automatically when the dialog element loses the focus.

Parameters

Name/Data Type	Explanation
HANDLE OF dialog	Input The dialog in which all linked variables are to be refreshed.
Response (I4)	Output Natural error (if applicable).

Example:

```
#MYVARI := 'Refreshed'
PROCESS GUI ACTION REFRESH-LINKS WITH #DLG$WINDOW GIVING #RESPONSE
```

RESET-ATTRIBUTES Action

Description

Resets all attributes in the global attribute list to their default values. For the default values, see the section Attributes. It is recommended to use this action before creating dialog elements dynamically with the ADD action, for example, in an after open event handler.

Parameters

Name/Data Type	Explanation
Response (I4)	Output Natural error (if applicable).

Example:

```
/* After open event handler code
PROCESS GUI ACTION RESET-ATTRIBUTES GIVING #RESPONSE
PROCESS GUI ACTION ADD...
```


SAVE-LAYOUT Action

Description

Saves the specified dialog's current control bar layout for the current user. The control bars include the newer tool-bar and status-bar controls, but not the traditional tool bar and status bar available in earlier Natural versions. The layout includes the bar type (dockable or fixed), location, position, and the bar's visible and enabled state.

The information is currently stored in the registry under:

```
\\HKEY_CURRENT_USER\Software\Software AG\Natural Applications\<profile name>\<section name>
```

where <profile name> and <section name> are described in the table below. However, the base location (or whether the information is stored in the registry at all) may change for future Natural versions.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Handle of the dialog for which the control bar layout is to be saved.
Profile name (A253)	Input (optional parameter) The name of the profile to which the information is to be saved. If this parameter is not specified, the name of the library containing the dialog is used.
Section name (A253)	Input (optional parameter) The name of the profile section to which the information is to be saved. If this parameter is not specified, the name of the dialog is used.

Example:

```
PROCESS GUI ACTION LOAD-LAYOUT WITH #DLG$WINDOW GIVING #RESPONSE
```

SET-FOCUS Action

Description

Selects a dialog element so that the end user can enter data with the keyboard. You can use this action with handles of the following dialog elements: bitmap control, edit area control, input-field control, list box control, push button control, radio button control, scroll bar control, selection-box control, toggle button control. You can also use this action with a dialog's window handle, for example, with "#DLG$WINDOW".

Note:
It is not recommended to execute the SET-FOCUS action from the leave event, because the leave event implies that the focus is to be set to yet another part of the application.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Any dialog element of the above-mentioned type that is supposed to get the focus.
Response (I4)	Output Natural error (if applicable).

Example:

```
#INPUT := #IF-1
PROCESS GUI ACTION SET-FOCUS WITH #INPUT GIVING #RESPONSE
```


SET-TABS Action

Description

Sets tabulator stops in a list box control even if the font in the list box control is proportional. The tabulator stops apply to the STRING attribute in any item of the list box control.

Parameters

Name/Data Type	Explanation
HANDLE OF LISTBOX	Input Handle of the list box control.
Tabstop (I4)	Input The tabstop is measured in units of one quarter of the system font's average width. You can enter this parameter any number of times (in ascending order). In the list box items' STRING, you determine the place of the tabstop by inserting the HORIZONTAL-TAB character (defined in the local data area NGULKEY1). Inserting this character will move the following STRING text to the next tabstop.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION SET-TABS WITH #LB-1 50 100 150 GIVING #RESPONSE
COMPRESS 'HELLO' HORIZONTAL-TAB 'HUGO' TO #STRING
PROCESS GUI ACTION ADD-ITEMS WITH #LB-1 1 #STRING
```


SYSTEM-GET-NATIVE-HANDLE Action

Description

Queries the native handle of a dialog element in terms of the windowing system. You can use this native handle for external function calls.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input The handle of the dialog element in Natural terms.
Native handle (I4)	Output The window handle of the dialog element in terms of the windowing system. For font controls, the native font handle is returned. For menu bars, menu items and submenu controls, the native menu handle is returned. For bitmap controls, the native bitmap handle is returned. For all other dialogs and dialog elements, the native window handle is returned.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION SYSTEM-GET-NATIVE-HANDLE WITH #PB-1 #NATIVEHANDLE GIVING
#RESPONSE
#IF-1.STRING := #NATIVEHANDLE  /* Display the handle value
```


SYSTEM-PRINTER-SETUP Action

Description

Invokes a dialog box that allows the end user to assign a physical printer to a logical device name. You can assign this logical device name to a report using the DEFINE PRINTER statement.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input The parent dialog element of the printer assignment dialog box.
Logical device (A8)	Input/Output The logical device name to which the printer is assigned. Possible values: "LPT1" to "LPT31", "lpt1" to "lpt31".
Commit (L)	Output (optional parameter) Returns TRUE if the user closed the dialog box via the ’OK’ push button, or FALSE if the dialog was cancelled or if an error occurred.
Response (I4)	Output Natural error (if applicable).

Note:
The first two parameters are mandatory.

Example:

```
PROCESS GUI ACTION SYSTEM-PRINTER-SETUP WITH #PB-1 ’LPT1’ GIVING #RESPONSE
```

TABLE-DELETE-ROW Action

Description

Deletes a row in a table. To do so, one of the table control’s STYLE attribute values must be "e". Specifying the value "0" in the Row index parameter is useful in combination with the delete-row event because the ROW attribute then contains the index of the row in which the end user has caused the most recent event.

Parameters

Name/Data Type	Explanation
HANDLE OF TABLE	Input Specifies a table control.
Row Index (I4)	Input Specifies the number of the row to be deleted. If you specify "0", the value of the table’s ROW attribute is taken as the number of the row to be deleted. If you specify "-1", the last row in the table will be deleted. If you specify "1", the first row in the table will be deleted.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION TABLE-DELETE-ROW WITH #TBL-1 2 GIVING #RESPONSE
...
PROCESS GUI ACTION TABLE-REFRESH WITH #TBL-1 GIVING #RESPONSE /* Recommended
```


TABLE-FIND-FIELD Action

Description

Finds the column with the specified title.

Parameters

Name/Data Type	Explanation
HANDLE OF TABLE	Input Specifies a table control.
Field name (A253)	Input The column title to be found.
Index (I4)	Output The index of the column where the title was found.
Response (I4)	Output Natural error (if applicable).

Example:

```
   PROCESS GUI ACTION TABLE-FIND-FIELD WITH #TBL-1 FLD1 #IDX GIVING #RESPONSE
```

TABLE-GET-SELECTION Action

Description

Retrieves the column and row indices of the selected cells in a table control from the beginning of the first selected cell to the end of the last selected cell.

Parameters

Name/Data Type	Explanation
HANDLE OF TABLE	Input Specifies a table control.
Column from (I4)	Output First selected ...
Row from (I4)	Output ...cell.
Column to (I4)	Output Last selected ...
Row to (I4)	Output ...cell.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION TABLE-GET-SELECTION WITH #TBL-1 #COLFIRST #ROWFIRST #COLLAST
#ROWLAST GIVING #RESPONSE
```


TABLE-INQUIRE-CELL Action

Description

enables you to inquire which cell is at a particular pixel offset within the table. This is particularly useful for determining the cell for which a context menu was invoked, in which case the position specified should be that returned by the INQ-CLICKPOSITION action used within the context menu's BEFORE-OPEN event.

Parameters

Name/Data Type	Explanation
HANDLE OF TABLE	Input Specifies a table control.
X-Position (I4)	Input Specifies the x-axis position in pixels relative to the top-left hand corner of the table.
Y-Position (I4)	Input Specifies the y-axis position in pixels relative to the top-left hand corner of the table.
Row Index (I4)	Output The row corresponding to the specified y-axis position, or 0 if outside the range of any row.
Column Index (I4)	Output The column corresponding to the specified x-axis position, or 0 if outside the range of any column.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION TABLE-INQUIRE-CELL WITH #TBL-1 #X-POSITION #Y-POSITION #ROW #COLUMN
GIVING #RESPONSE
```

TABLE-INQUIRE-ROW Action

Description

enables you to inquire the value in each cell of a single row. The values will update the input parameters you specify. Later at runtime, you can retrieve the updated parameter values. Specifying the value "0" in the "Row Index" parameter is useful in combination with the insert-row event because the ROW attribute then contains the index of the row in which the end user has caused the most recent event.

Parameters

Name/Data Type	Explanation
HANDLE OF TABLE	Input Specifies a table control.
Row Index (I4)	Input Specifies the number of the row to be inquired. If you specify "0", the value of the table's ROW attribute is taken as the number of the row to be inquired. If you specify "-1", the last row in the table will be inquired. If you specify "1", the first row in the table will be inquired.
List of parameters	Input You can either specify one parameter for each column of the row, or you can specify less parameters than there are columns. If you specify more parameters than there are columns, an error message is returned.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION TABLE-INQUIRE-ROW WITH #TBL-1 3 #P1 #P2 #P3 #P4
GIVING #RESPONSE
```


TABLE-INSERT-ROW Action

Description

Inserts a row into a table. To do so, one of the table control's STYLE attribute values must be "e". Specifying the value "0" in the "Row Index" parameter is useful in combination with the insert-row event because the ROW attribute then contains the index of the row in which the end user has caused the most recent event.

Parameters

Name/Data Type	Explanation
HANDLE OF TABLE	Input Specifies a table control.
Row Index (I4)	Input Specifies the number of the row before which the new row is to be inserted. If you specify "0", the value of the table's ROW attribute is taken as the number of the row before which the new row is to be inserted. If you specify "-1", the new row will be appended to the lat row in the table. If you specify "1", the row to be inserted will become the first row in the table.
List of parameters	Input You can either specify one parameter for each column of the row, or you can specify less parameters than there are columns. If you specify more parameters than there are columns, an error message is returned.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION TABLE-INSERT-ROW WITH #TBL-1 3 #P1 #P2 #P3 #P4
GIVING #RESPONSE
PROCESS GUI ACTION TABLE-REFRESH WITH #TBL-1 GIVING #RESPONSE /* Recommended
```


TABLE-REFRESH Action

Description

Refreshes the contents of a table control.

Parameters

Name/Data Type	Explanation
HANDLE OF TABLE	Input Specifies a table control.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION TABLE-REFRESH WITH #TBL-1 GIVING #RESPONSE
```


TABLE-SET-SELECTION Action

Description

Selects a rectangular range of cells in a table control.

Parameters

Name/Data Type	Explanation
HANDLE OF TABLE	Input Specifies a table control.
Column from (I4)	Input First selected ...
Row from (I4)	Input ...cell.
Column to (I4)	Input Last selected ...
Row to (I4)	Input ...cell. If you specify "0", the selection will extend to the last cell in the table.
Response (I4)	Output Natural error (if applicable).

Example:

```
PROCESS GUI ACTION TABLE-SET-SELECTION WITH #TBL-1 #COLFIRST #ROWFIRST #COLLAST
#ROWLAST GIVING #RESPONSE
```


TEXT-GET-EXTENT Action

Description

Returns the size of a text if a certain font is chosen. You can use this action, for example, to find out the best font that still has an acceptable width and height.

Parameters

Name/Data Type	Explanation
HANDLE OF FONT	Input Specifies a font control.
Text (A253)	Input Your text.
Width (I4)	Output Width in pixels.
Height (I4)	Output Height in pixels.
Response (I4)	Output Natural error (if applicable).

Example:

```
/*Check the size a font will require
#TEXT := 'Check the size'
PROCESS GUI ACTION TEXT-GET-EXTENT WITH FONT-HANDLE #TEXT #WIDTH #HEIGHT
GIVING #RESPONSE
```


UPDATE-COMMAND-STATUS Action

Description

A dialog automatically receives command-status events (unless suppressed) during idle processing when the status of one or more commands (signals, or menu items or tool bar items without a SAME-AS attribute) need to be updated. This action, however, allows the same updating logic to be triggered manually at any time, in addition to the automatic idle-time update.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Handle of the dialog or dialog element whose status is to be updated, or NULL-HANDLE. If a dialog handle is specified, all commands belonging to the specified dialog (excluding submenus and context menus) are included in the updating process. If a dialog element handle is specified, all commands belonging to the specified dialog element are included in the updating process.If NULL-HANDLE is specified, all commands in all dialogs (excluding submenus and context menus) are included in the updating process, as is the case for the implicit idle-time update.

Example:

```
PROCESS GUI ACTION UPDATE-COMMAND-STATUS WITH NULL-HANDLE GIVING #RESPONSE
```

VALIDATE Action

Description

Validates a handle variable. A handle variable is considered valid if it contains either the value NULL-HANDLE or a handle to an existing dialog or dialog element.

Parameters

Name/Data Type	Explanation
HANDLE OF GUI	Input Any dialog or dialog element.
Response (I4)	Output Natural error (if applicable), or 0 if handle variable is valid.

Example:

```
PROCESS GUI ACTION VALIDATE WITH #PB-1 GIVING #RESPONSE
```

